

Unsupervised Context Detection using Wireless Signals

Dinh Phung[†], Brett Adams[†], Svetha Venkatesh[†] and Mohan Kumar[‡]

[†]*Department of Computing, Curtin University of Technology,
GPO Box U1987, Perth, WA 6845, Australia.*

[‡]*Department of Computer Science and Engineering
The University of Texas at Arlington, Box 19015, Arlington TX 76019, USA.*

Abstract

Sensing context plays an important role in many pervasive and mobile computing applications. Continuing from previous work [33], we present an unsupervised framework for extracting user context in indoor environments with existing wireless infrastructures. Our novel approach casts context detection into an incremental, unsupervised clustering setting. Using WiFi observations consisting of access point identification and signal strengths freely available in office or public spaces, we adapt a density-based clustering technique to recover basic forms of user contexts that include user motion state and significant places the user visits from time to time. High-level user context, termed rhythms, comprising sequences of significant places are derived from the above low-level context by employing probabilistic clustering techniques, latent Dirichlet allocation and its n-gram temporal extension. These user contexts can enable a wide range of context-ware application services. Experimental results with real data in comparison with existing methods are presented to validate the proposed approach. Our motion classification algorithm operates in real-time, and achieves 10% improvement over an existing method; significant locations are detected with over 90% accuracy and near perfect cluster purity. Richer indoor context and meaningful rhythms, such as typical daily routines or meeting patterns, are also inferred automatically from collected raw WiFi signals.

Key words: context modeling, spatio-temporal rhythm extraction, probabilistic topic models, hidden Markov models, unsupervised learning, wireless signals

1. Introduction

The proliferation of small computing mobile devices, such as smartphones, iPods, eEPCs, ultraportable computers has brought new opportunities and challenges for ubiquitous computing [41]. On one hand, a more prevalent trend is that these devices are rapidly equipped with sensing capabilities: GPS receivers, WiFi, Bluetooth transceivers, and many more. On the other hand, mobile computing has increasingly broadened computing perspective in everyday user's context, being, at least, either situational or interactional. Three useful contexts are motion state, location and movement patterns over time and space. Location often correlates with certain activities or roles [30], and aspects of the user's physical states, such as motion, are also indicative of activity and the user's affordances (e.g. interruptibility). This information can drive applications at many levels, from automated battery management to assistive systems (e.g., for the visually

Email addresses: d.phung, b.adams, s.venkatesh@curtin.edu.au, mkumar@uta.edu (Dinh Phung[†], Brett Adams[†], Svetha Venkatesh[†] and Mohan Kumar[‡]).

impaired). Device convergence has made available a number of sensing and communication technologies, including Bluetooth, WiFi and GPS, for extracting these elements of context. Our early work has considered the problem of context extraction in an outdoor setting using GPS [34,2]. In this paper, we design, develop and evaluate novel, high accuracy mechanisms for extracting user context in indoor environments. User motion level and significant locations, defined as places where a user spends time, are extracted from raw WiFi signals in a timely, unsupervised, and accurate manner, in existing wireless network infrastructures. We further provide methods on how this fundamental information can be used to discover such higher-level contexts as the user’s daily routines or *rhythms* in an unsupervised setting.

Much work has been done in the use of GPS to derive representations of significant locations in outdoor situations. There has also been significant activity in localization from signatures that penetrate or originate indoors, such as WiFi, GSM, and Bluetooth. Work aimed at characterizing the physical state of a user has tended to make use of sensors that aren’t as readily available as ambient radio signatures, such as thermometers, galvanic sensors and accelerometers. A brief review of relevant work is provided in Section 2. In [38] location and orientation estimations based on Bayesian filtering of received signal strength (RSS) justifies the use of WiFi signals for extracting location context. The Locadio positioning system of [23] uses WiFi signals to infer whether or not a user is moving based on the variance of signal of the strongest WiFi access point (WAP) within a short window, achieving an accuracy of 87% after a post-processing step with a 2-stage hidden Markov model (HMM). Often, the noisy, sparse nature of WiFi signatures renders models based on simple Gaussian assumption become problematic in this case. Additional shortcoming of Locadio is the requirement for training. Moreover, in the original setting of Locadio [23], prediction is made with a latency of 20 seconds, which disqualifies the approach from real-time applications, such as navigation assistance for the visually impaired.

Build upon our preliminary work in [33], instead of viewing motion state detection as a supervised classification problem, we cast it as an unsupervised and incremental clustering problem. A window of consecutive WiFi signatures observed from the same location, when the user is still, are likely to be similar, and thus is more likely to form a dense cluster as opposed to those when the user is moving. Similarly, if WiFi signals observed during a user’s daily life are collated, locations where the user spends time repeatedly, for example at their desk at work, will also emerge from a clustering process. We define a measure of distance between two WiFi observations appropriate to their characteristics, notably allowing for missing data from the vectors of WAP signal strength. We use a density-based technique, DBSCAN [13] and its incremental version [12], to recover user motion level and significant locations. Use of incremental DBSCAN allows for motion level classification with latency under 2s, which can potentially be suitable for many real-time applications. We conduct comprehensive experiments to compare Locadio method with our density-based approach for detecting user state. We achieve up to 95% accuracy, an improvement of 10% over Locadio’s approach in [23], suggesting that our method is more robust despite noisy and incomplete WiFi data. We experiment with detection of significant location, using pre-filtering to remove observations when the user is moving, resulting in an accuracy of above 91% with almost perfect clustering purity (98%).

To further motivate the extraction of motion state and significant locations, we also present a technique for discovering user indoor movement behaviors over time, termed *rhythms*. It has been shown that travel episodes often correspond to hidden agendas or ‘social projects’ [8], and we posit that a similar situation occurs at finer resolutions, say, within the office. Discovery of these rhythms offers potentially rich information about user intent and activity. We adapt a probabilistic graphical model, Latent Dirichlet Allocation (LDA)[5], for this task. LDA is an unsupervised probabilistic clustering technique used to discover latent topics from bags of words in text by finding co-occurrences of words in documents. Here, significant locations and their observed times are extracted and are mapped to words. These are then collated over a day and become analogous to a document. The latent topics discovered by LDA in this way are interpreted as user rhythms. We experiment with the discovery of rhythms for a user over the course of a one month period.¹ While interpretable and meaningful office patterns can be clustered, LDA does not truly model the temporal

¹ It is worth noting from the perspective of assistive systems that the incidence of strict routines is even higher among the visually impaired, presumably due in part to the desire to decrease the number of variables that might induce danger or inconvenience for themselves or others, making rhythms more compelling in this application domain.

information between landmarks due to its strict ‘bag-of-word’ assumption. To this end, we extend our previous work using LDA in [33] to employ an n-gram extension of LDA known as n-gram topic model (NLDA) [40] and show that richer trajectory patterns can be automatically discovered such as typical daily routines or research meeting patterns. In all cases, the results from the hidden Markov model are also presented as a baseline performance for comparison.

First novel contributions in this paper, which were earlier developed in [33] are two-fold: the development of unsupervised and incremental algorithms for high accuracy motion; and the real-time detection of significant places visited using WiFi signals. *In addition to [33], a significant novel contribution unique to this paper is the extraction of rhythms using rich Bayesian probabilistic models that explicitly capture temporal and spatial information in a sensor agnostic manner in an indoor setting.* The ability to infer a mobile user’s context is a vital and foundational component for a broad array of pervasive computing applications. Our investigations enable both richer representation and more accurate extractions of different aspects of context, and hence the outcomes of the research work presented in this paper can potentially be valuable to various applications. The two-fold contributions can serve as a basis for both annotation and prediction at a number of levels of the services stack. Importantly, the absence of a requirement for calibration and use of existing infrastructure make for a low barrier to deployment. The applications include, context-sensitive device resource and interface management, semi-automatic calendaring, personal life logs and collaboration tools, personalized push-information such as advertising, and navigation assistance for the visually impaired. In a shared context, this information can aid market research, surveillance and urban planning.

The remainder of the paper is structured as follows. Section 2 discusses related work in context extraction, with a particular emphasis on indoor localization from WiFi signatures. Section 3 discusses the envisaged operational setting of our algorithms for motion state classification, significant location extraction and rhythm discovery. Section 4 begins with a description of our experiment setup, and presents the results for each of the clustering tasks. We conclude with summary remarks in Section 5. Finally, the Appendix A contain further supplementary results for the experiments reported.

2. Related Work

Thanks to the development of GPS, several methods for detecting location-based activities and significant places visited have been developed in outdoor environments [10,45,1,25]. The work of [10] infers high-level human behaviors from low-level GPS traces using a variant of the Abstract Hidden Markov Model in [7] with an extension to handle continuous observation. [25] proposes a form of hierarchical conditional random field, incorporating domain knowledge of hierarchical spatial proximity structure, to learn daily activities and significant place from GPS data. Dynamic probabilistic models, being either generative [10] or discriminative [25], have been employed in these works to handle uncertainties in low-level signals and activity structures, where raw GPS readings are often mapped as observation at the bottom level of the model and high-level activities are modeled at higher levels. These approaches are, however, supervised and thus require an expensive parameter estimation phase and often do not support inference for growing data sets. Using non-probabilistic approaches, [45] and [1] employ density-based clustering to extract significant places from GPS. Table 1 provides a brief breakdown of related work by signal type, method, context discovered, and potential cost involved.

We focus primarily on work that recovers context with an indoor component in terms of location, proximity of others, and/or some definition of activity via a range of sensors with low-cost. The work of [9] develops a wearable system including potentially expensive video and audio sensors for extracting the events and scenes. HMMs are used to infer the events, however recent recognition techniques in video and audio are still unreliable, in addition to the computational burden in the training HMM models. Similar systems are also developed in [6] using mainly audio, accelerometers and WiFi in a supervised framework to infer context such as posture, location and activities. The main drawback with these systems is again the overload in computational power to process videos and audios, and thus, not suitable for real-time demands when operating on mobile devices with limited computing resources. Additionally, they require collecting training data to for training classifiers and thus are different from ours where no training data is required. [11] uses

	Signal	Context	Work	Method	Cost
Indoor	WiFi	proximity, neighborhood	[26]	physical proximity model	low
	WiFi	location	[4]	k -nearest neighbors	low
	WiFi	location	[24,36,44,42]	naive Bayes model	low
	WiFi	user motion state	[23]	variance statistics on RSS and HMM	low
	RFID	activity modeling	[32]	dynamic Bayesian network	low/medium
	audio, accelerometer, WiFi, videos	posture, location, speech, activities	[6]	decision tree, Bayesian classifiers	medium/high
	Powerline	location	[31]	naive Bayes model	low/medium
Outdoor	GPS	activity modeling	[10]	dynamic Bayesian network	low/medium
	GPS	significant places	[3]	clustering with heuristics thresholds	low/medium
	GPS	significant places	[45]	density-based clustering with time	low/medium
	GPS	significant places	[1,2]	density-based clustering	low/medium
	GPS	activity modeling and recognition	[25]	hierarchical conditional random fields	low/medium
	GPS	significant places, movement patterns	[19]	hidden Markov models	low/medium
	GPS	significant places, social rhythms	[2]	density-based clustering	low/medium
Hybrid	visual, aural	events, scenes	[9]	short/long time-scale HMM	high
	GSM, cell tower ids	eigenbehaviours (zero-order patterns)	[11]	singular value decomposition (SVD)	low
	GSM, bluetooth	routine discovery	[14]	probabilistic latent semantic analysis (PLSA)	low/medium
	accelerometer	activity recognition	[21]	naive Bayes model	medium

Table 1

A brief survey on context extraction literature and methods used. Note that none of these approaches address the incremental and real-time detection of context, except [2].

GSM positioning and Bluetooth proximity to extract the repeated activities of individuals and community patterns by extracting the principal eigen-behaviors from the eigenvalues of the day (row) vs time (column) matrix, in which an entry indicates if the user was at that spatial temporal location. However, GSM is too coarse for use in indoor activities. Further they do not extract rhythms. By integrating the RFID tag for each equipment with meaningful description [32], the context is extracted directly from the description and then higher-order activities are inferred using the same model in [10].

In [37], a Nash H-learning mechanism is used to predict user mobilities for efficient resource management. To complement location-based activity recognition, [21] propose a new approach to extract activity patterns using accelerometers. Applying the original LDA model with variational inference [5], [21] constructs a dictionary of word and a corpus of documents in which each word is an activity and a document consists of all words during particular day and leverage the daily patterns from normal activities collected over 16-day period using accelerometer. Naive Bayes classifier is further used for recognizing the normal activities such as sitting, walking, driving etc. The work of [34] proposes a novel model LSDA, an extended version of n-gram model [40], to extract the socially hidden rhythms of a user using GPS traces. The corpus of words

and documents is generated from GPS data [1] in which word is mapped to $\langle \text{time}, \text{significant place} \rangle$ and a document consists of all words in each day. The advantage of LSDA is its ability to map consecutive locations as a n-gram of words and thus in its ability to extract meaningful social rhythms.

Regarding the task of high-order social movement pattern discovery, little work has been found, mainly in an outdoor setting [19,2,11]. These works are often limited to ‘zero-order’ patterns, i.e., inferred rhythmic repetitions from a fixed location rather than between locations. For example, the extraction of rare, frequent, timed or optional rhythms in [2] is limited to being at a particular latitude-longitude landmark. [19] experiments with first-order patterns using the HMM but concluded that it is too strict. [43] employed two-level dynamic Bayesian networks to infer users’s activities from WiFi signals similar to the way our early work [28] used the hierarchical hidden Markov models for activity recognition with visual sensors. However, these approaches are supervised and particularly computational expensive, render them unsuitable for mobile platforms. The work of [11] applied singular valued decomposition (SVD) on the term-by-document matrix to extract principal eigenvectors interpreted as eigenbehaviors, analogous to rare and frequent rhythms extracted in [2]. In addition, this approach is not scalable since SVD computation is $O(n^3)$ complexity where n is the number of data points. [14] also look to characterize routines present in the Reality Mining dataset. In contrast to [11], they frame the problem in a probabilistic setting, and discover routines at the level of a day in both location and proximity using Probabilistic Latent Semantic Analysis (PLSA). Four codes are used to map cell tower IDs to locations, Home, Work, Other and No Signal, in a 30 minute period. A sequence of 3 locations are then added to a coarse time of day zone (e.g., 7–9AM) to yield a ‘word’ in a day’s ‘document.’ Similarly, proximity is encoded using two device IDs and a coarse timeslot. The topics discovered by PLSA are ranked, and manual observation yields behaviours such as “going to work early” or “meeting for lunch.” Our work differs in a number of significant ways: location vocabulary distinguishes between each and every significant place a user visits, allowing for discovery of specific sequences; and the n-gram topic model explicitly models sequence information, and is thus data driven.

Understanding context and providing context-aware application services are critical to dynamic pervasive environments. Context continues to be a topic of research focus as context data and their associated sources exhibit dynamism. Henrickson and Induslka [20] discuss shortfalls of context modeling and reasoning with ontologies for understanding context. Nicklas et al. [29] investigate the use of hybrid reasoning to augment the NEXUS framework. Existing work has considered dynamic models applied on either WiFi or accelerometer data to derive context in indoor environments. These models are, however, often strictly supervised, and limited efforts has been made to extract context with topic models on accelerometer and GPS data, but none of these have been applied to WiFi data. Thus there is a requirement to produce techniques that work in an unsupervised manner on noisy WiFi data to extract context.

3. A System for Context Detection

This section begins with an overview of the system, together with examples of its envisaged setting and uses by way of motivation. Separate sections are then devoted to motion classification, significant location extraction, and rhythm detection, respectively.

3.1. System Overview

The envisaged setting for the algorithms detailed below considers any mobile device equipped with a WiFi receiver. The system would typically run as a background process, making context information available as it is extracted. The input to the system consists of time-stamped vectors of received signal strength, each identified by the broadcasting WAP: $\{\text{time}, \text{WAP id}, \text{RSS}\}$. This is depicted on the left of Figure 1. Motion classification is performed with the arrival of each new WiFi sample (after an initial, small start-up latency period) and is made available immediately; for example, in indoor navigating, points at which the user becomes stationary are candidates for making reliable location prediction or issuing new information.

Referring to Figure 1, motion classification is an output of the *Context Detection* box. If the device also has the ability to store a historical record of WiFi samples, significant location discovery can also begin

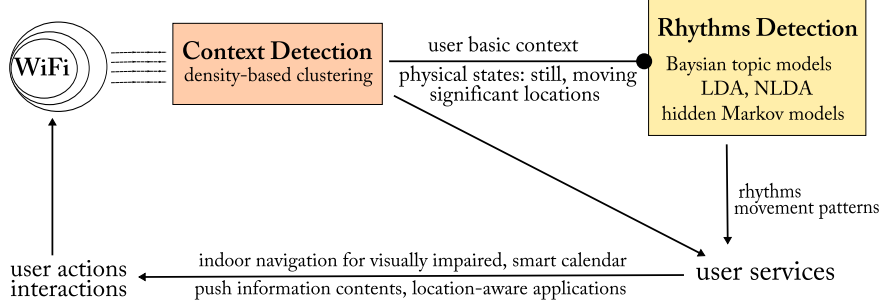


Figure 1. System overview: unsupervised context detection from raw WiFi signals.

immediately, resulting in a growing representation of where the user is spending time. E.g., in the course of a normal work day, a handful of locations might be discovered corresponding to the office, cafe, library and a colleague’s room. Significant locations are also depicted as outputs of context detection in Figure 1. These locations can be used as annotations to associate activities or media items (e.g., this is the set of applications you run at this location; you took these photos in the same place). If appropriate, labeling these locations meaningfully would be performed as a secondary activity, e.g. via active learning prompts, user-derived sources such as a calendar, or centrally-sourced such as beacon databases or pre-calibrated maps [2]. Finally, at the coarser resolution of days and weeks, rhythm detection becomes appropriate. As depicted on the right side of Figure 1, the rhythm detector accepts the user’s history of time-stamped landmarks and yields patterns of behavior in the user’s whereabouts. E.g., discovered rhythms might correspond to: an average work day, involving the office and home; a work day that includes collaboration or shopping; and weekend routines that have little overlap. Rhythms, in addition to constituting a higher-order object for annotation (e.g., these photos were taken at work, but not an average work day), provide the basis for prediction. E.g., the user typically doesn’t appear at these locations over the weekend.

3.2. WiFi observation distance

As we desire to cluster WiFi observations, we require a measure of distance between two such observations. In theory, the relationship between RSS and distance for a given WAP is inverse squared, and at first glance, modeling these points according to a Gaussian distribution and then performing hypothesis testing on the concentration of these points may offer a straight solution. However, there are a number of factors that complicate this model in practice: RSS is attenuated by physical structures and other environmental factors, which result in relatively high signal variability. Moreover, measuring the distance of a pair of *sets* of WAPs is complicated by missing values from one observation to the next. This also renders clustering algorithms that utilize Gaussian properties (e.g., GMM) unsuitable. One advantage of density-based clustering approaches is a degree of freedom in the formulation of a suitable function of distance between two observations.

Let $\mathcal{X} = \{x_1, x_2, \dots, x_N\}$ be the set of all WAPs available. For a WiFi observation p , let P be a subset of $\{1, 2, \dots, N\}$ denoting the set of WAP indices observed and X_P be the actual set of WAPs. E.g., if $P = \{2, 5\}$ then $X_P = \{x_2, x_5\}$. Furthermore, we denote by $y_i^{(p)}$ the corresponding RSS reading in observation p from the source x_i . Given two WiFi observations p and q , denoting the common WAPs set by $C = P \cap Q$, the distance between them is defined as:

$$\text{dist}(p, q) = \sqrt{\frac{1}{|C|} \sum_{j \in C} \left[y_j^{(p)} - y_j^{(q)} \right]^2} \quad (1)$$

where $|C|$ denotes the cardinality of set C . By this distance measure, only shared signal strength from common WAPs between two observations is taken into account, any missing ones do not affect the distance.

In practice, the spatial proximity of two WiFi observations affects the difference between X_p and X_q . Let $\eta = |C| / \max(|P|, |Q|)$ and using a threshold $\eta_0 \in [0, 1]$, the distance between p and q is adjusted to:

$$\text{dist}(p, q) = \begin{cases} \sqrt{\frac{1}{|C|} \sum_{j \in C} [y_j^{(p)} - y_j^{(q)}]^2} & \text{if } \eta > \eta_0 \\ +\infty & \text{otherwise} \end{cases} \quad (2)$$

Intuitively, the introduction of η_0 is to account for the case when the difference of observed WAPs in two observations is too large. For example, with $\eta_0 = 0.5$, any pair of observations that share less than half their WAPs in common will be set to be totally different ($+\infty$).

3.3. Motion classification

As mentioned earlier, the key observation used to infer about a user’s motion state is the level of ‘denseness’ or ‘connectedness’ of WiFi observations accumulated within a short time interval acquired incrementally in real-time. A good fit for this task is DBSCAN [13], a density-based clustering algorithm, with the additional advantage of being non-parametric in number of clusters.

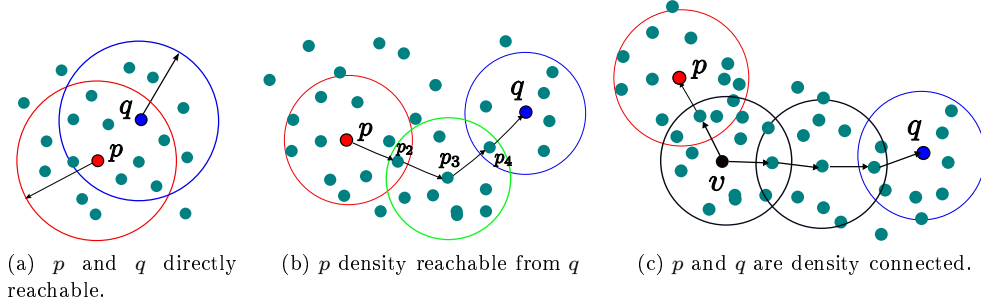


Figure 2. Illustration of three key concepts in density-based clustering: (a) directly density reachable, (b) density reachable, and (c) density connected [13].

DBSCAN develops three concepts that are naturally relevant to our problem: *directly density reachable*, *density reachable* and *density-connected*. It requires a pair of parameters (ϵ, D) (which may be inferred automatically from the data) where ϵ is a radius around point p to form its neighboring set $N(p|\epsilon)$ and D serves as a threshold to test if two points p and q are *directly density reachable*. A point q is then called *density reachable* from p if there is a sequence of points $\{p_1, \dots, p_l\}$ that connects them, i.e., $p = p_1 \rightarrow p_2 \dots \rightarrow p_l = q$, where p_i is directly density reachable to p_{i+1} ; and finally p is called density connected to q if there exists a point v that is density reachable from both p and q . These concepts are depicted in Figure 2. DBSCAN then seeks to form clusters that are maximal in density-connectedness. Incremental DBSCAN [12] also uses the above fundamental density concepts but operates in an online manner. Assuming that all points prior to the arrival of a new point have been clustered, the essential step is the INSERT operator, which updates the points effected by the newly arrived point. A DELETE operation can also be similarly performed to remove stale points.

To determine whether the user is static or moving, we examine the similarity of WiFi observations within a window. If the user is static, a regular ‘cluster’ will be returned, and if the user is moving, no cluster should be formed because of the variability in WiFi signal strength or visible access points.

The batch or off-line approach is restricted by window size (Alg. 1). E.g., if the window size is 20 seconds, a result can be obtained only after each 20 second window is processed, leading to clear real time limitations. This can be overcome by an overlapping window approach that employs Incremental DBSCAN. As each new overlapping window is introduced, new WiFi observations are added, and outdated WiFi observations are removed. As each WiFi observation p is introduced, there are three possibilities: (1) Noise: p is considered noise, (2) Creation: p and some previous noise points form a new cluster, or (3) Absorption: p is absorbed into the existing cluster (Alg. 2). There are two cases when a point p is to be removed: (1) Removal: p ’s neighbors are decreased and the existing cluster may disappear, or (2) Reduction: p ’s neighbors are decreased but cluster status is unchanged (see Alg. 2). To take advantage of the existence of at most one cluster, the algorithm terminates as soon as a cluster is found.

Algorithm 1 Motion state detection.

Input: current window of WiFi observations

```
1: mark all points inside the window as unclassified.
2: for each observation  $p$  in current window do
3:   if no cluster found and  $p$  is unclassified then
4:     if number of neighbors  $|N(p|\epsilon)| \geq D$  then
5:        $p$  and its neighboring points form a cluster.
6:     else
7:        $p$  is noise point
8:     end if
9:   end if
10: end for
```

Output: motion state: ‘moving’ if no cluster found or cluster size is smaller a threshold δ (discussed more in the texts); or ‘static’ otherwise.

Recall that DBSCAN requires two parameters: the neighborhood radius ϵ and number of neighbors D . Ester *et. al.* [13] propose a simple but efficient heuristic to determine ϵ and D in terms of the “thinnest” cluster in the database. Let k -dist be the distance from each point p to k -th nearest neighbor q of p . A sorted k -distance graph is produced by sorting all the points in descending value of k -dist. A good threshold is suggested to be empirically chosen as the point where there is a rapid change in the sorted k -distance graph. In Figure 6a we empirically show a threshold that intuitively gives a good balance on this criteria. If the number of observations assigned to a particular cluster within a window is greater than a threshold δ , the state is deemed to be static. For a window with N observations, D is calculated as $D \approx \delta \times N$.

3.4. Significant location discovery

The notion of significant or *meaningful* locations has been discussed in the literature from a range of perspectives. For our purposes it is sufficient to define a significant location as a place where a user spends time. This in turn requires that the user be repeatedly *stationary* at a location.

Hence, in our setting, this involves a two step process. Firstly, the user’s motion level is classified over a window of time, and a cluster is formed if they are inferred to be still. If so, the observations that belong to the static cluster are extracted. For each WAP id in these observations, the average signal strength is computed. This is termed the “average observation” for the given window. In the second step, we use the original version of Incremental DBSCAN presented in [12] to cluster this new average observation. As each WiFi observation p is introduced, there are four possibilities: (1) Noise: p is a noise point, (2) Creation: p and some previous noise points form a new cluster, (3) Absorption: p is absorbed in the existing cluster, or (4) Merge: several clusters and noise neighbors are merged to form a cluster. Parameters are determined using the k -distance criteria referred to above, barring the parameter D , which is directly input from the system.

3.5. User rhythm detection

While the notion of significant locations is useful for many applications, location alone is an insufficient index for others. Often, diverse activities may be folded into the same location, and are only differentiable when the context of behavior over time is considered. In a study of human mobility, [16] have demonstrated, at a coarse scale using logs from 100 thousand mobile phone users over a six-month period, that human trajectories show “a high degree of temporal and spatial regularity.” [22] note that, in addition to location, other social facets including routine are equally important. Below we detail an approach to detecting periodicity in time and location that arise from daily routines, termed *rhythms*.

Generative process for rhythm extraction. Assume a device has been logging WiFi signals and clustering these signals into significant locations at the same time. Thus, unfolding in time and space we acquire

Algorithm 2 Incremental INSERT and DELETE operations.

Input: new point p (to INSERT) and existing data points

```
1: if number of neighbors  $|N(p|\epsilon)| \geq D$  then  
2:    $p$  belongs to the existing cluster or form a new one.  
3: end if  
4: for each neighboring point  $q$  of  $p$  do  
5:   update neighboring set of  $q$  by adding  $p$   
6:   if  $|N(q|\epsilon)| \geq D$  then  
7:      $q$  belongs to the existing cluster or form a new one.  
8:   end if  
9: end for
```

Output: updated clustering result and motion state.

Input: existing point p (to DELETE) and other data points

```
1: for each neighbor point  $q$  of  $p$  do  
2:   update neighboring set of  $q$  by subtracting  $p$   
3:   if  $|N(q|\epsilon)| < D$  then  
4:     update cluster status  
5:   end if  
6: end for
```

Output: updated clustering result and motion state.

a trajectory of movements $\mathbf{w} = \{w_1, w_2, \dots, w_N\}$ where each w_i is called a *spatio-temporal word* consisting of a pair $\langle \text{time}, \text{location} \rangle$ that the user has visited over time. For example, in Figure 10, ‘1730student’ is a spatio-temporal word indicating that the user (the student) was at his desk at 17:30; likewise ‘1030advisorY’ means he was in advisor Y’s office at 10:30am. By collecting WiFi data on a day-by-day basis over D days a corpus $\{\mathbf{w}^1, \dots, \mathbf{w}^D\}$ of trajectories is acquired.

Motivated by the results of [16] and [22], we treat a rhythm as a ‘hidden’ agenda or timetable that *generatively* gives rise to repetitive co-occurrences of visitations of significant locations observed over time (cf. Figure 3 and 4). This can be translated into the co-occurrence of words under the spatio-temporal word representation of WiFi trajectories presented earlier. For example, a rhythm of ‘meet with advisor at 11:00am, lunch at noon, then attend seminar at 1:00pm’ will assign high probability to the words ‘1100advisor’, ‘1200lunch’ and ‘1300seminar’. A helpful comparison is the task of topic modelling for natural language: terms such as ‘washington’, ‘speech’ and ‘budget’ are analogous to our derived spatiotemoral words, and the obtained topic, in this case ‘politics,’ is analogous to a rhythm. The generative process of rhythms will now be examined in detail.

Rhythm detection as a statistical inference problem. In our previous work [33] we applied a Bayesian topic model, known as latent Dirichlet allocation (LDA) [5], for rhythm detection, viewing this task as an inference problem over the hidden topic variables. LDA is an efficient method for document modeling, treating each document as a mixture of topics where each topic has a precise definition as a probability distribution over all words. Analogously, a user’s daily routines can be thought of as mixtures of different plans, agendas, or rhythms. To illustrate this process, an example is provided in Figure 3(a) and a formal probabilistic graphical model representation for LDA is shown in Figure 3, where WiFi observations are clustered and mapped to the spatio-temporal words mentioned earlier. These observations are collated each day to form a document, all days form a corpus, and the latent topics discovered from this corpus are the sought-after rhythms. The generative process is as follows. For each document d , a mixing topic proportion $\theta_d \sim \text{Dir}(\alpha)$ is sampled from a Dirichlet distribution parameterised by the hyperparameter α . Each word w in a document is generated by first sampling a topic z from a multinomial distribution $z \sim \text{Mult}(\theta)$, and then sampling $w \sim \text{Mult}(\phi_z)$ also from a multinomial distribution. Given the topic z , each $\phi_z \sim \text{Dir}(\beta)$ is sampled from a Dirichlet distribution parameterised by β . Thus, in LDA each document is a mixture of topics represented by θ_d and each topic is a distribution over all words represented by $\phi_{z,w} = \text{Pr}(w | z)$.

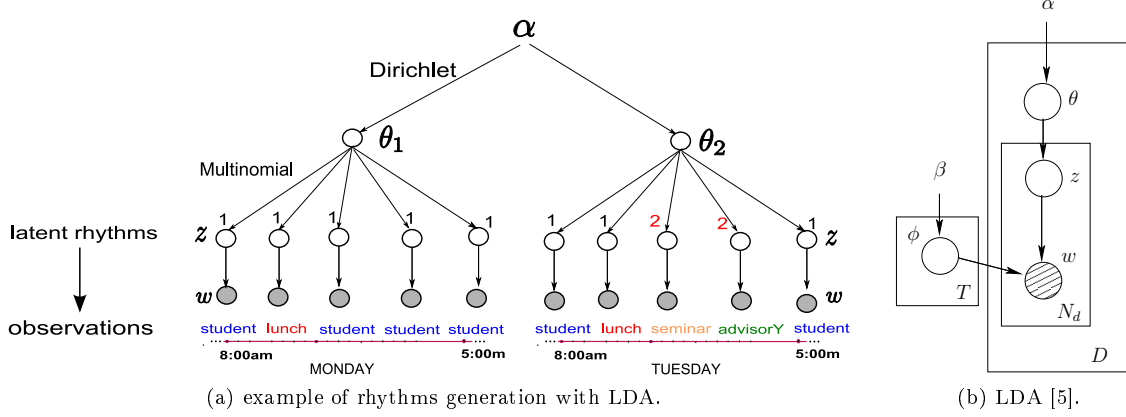


Figure 3. (a) A make-up example to illustrate generative process for rhythms when modelled as LDA which can be viewed as the expanded graphical model in (b) for $D = 2$ documents. In this example there are two rhythms $T = 2$. Monday represents a typical day (rhythm 1) for a student coming to work, have lunch, then go home. Thus, all the rhythms variables z is assigned to 1. On Tuesday, however, his typical routine is *mized* with going to the seminar and then meeting with the advisor in the afternoon (rhythms 2). (b) Formal graphical model representation of LDA in plate notation where T is the number of topics or rhythms, D is the number of documents of days in the corpus, N_d is the number of words contained in document d , w is an observed spatio-temporal word, z is the hidden topic or rhythm variable to be inferred (parameters θ, α and ϕ, β are discussed in the texts) .

The latter represents the rhythms (i.e., the topics) that we are interested to infer (cf. Algorithm 3).

Examining the graphical model of LDA in Figure 3 can easily reveal that each word w_i is independent of all other words given its immediate topic assignment z_i , which is known as the ‘bag-of-words’ assumption in language modelling. In other words, if we arbitrarily shuffle the order of words in a document, the LDA remains the same, it still assigns consistent joint probability over all variables. However, the temporal and spatial order is an important feature in the domain of data dealt with in this work. For example, from the perspective of the assistive systems for the visually impaired, strict movement routines from one place to another plays a crucial role in their daily routines due in part to the desire to decrease the chance of inducing danger or inconveniencing themselves or others. While LDA can potentially cluster meaningful rhythms, strict ordering of movements is lost.

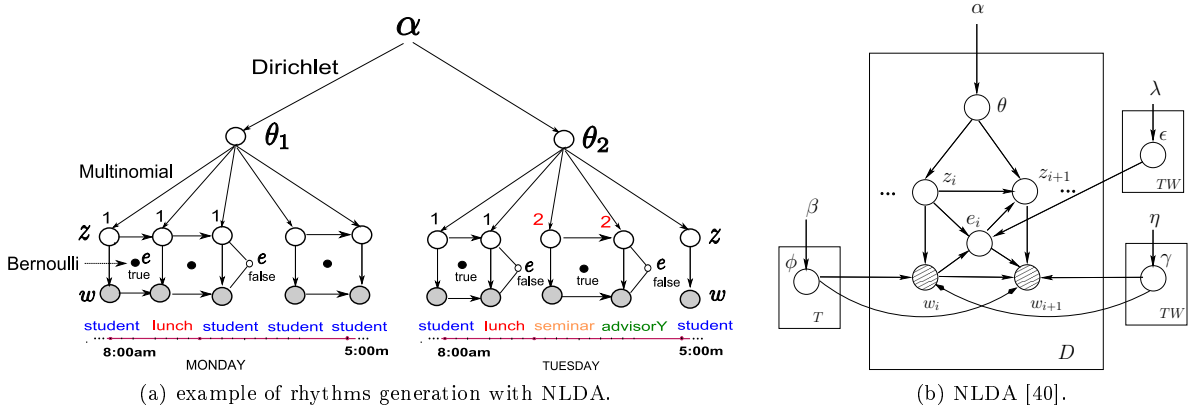


Figure 4. (a) A possible generative process for rhythms when modelled as NLDA with the same example shown in Figure 3. Again, it can be viewed as the expanded graphical model of the plate representation in (b). A (spatio-temporal) word w_i is no longer independent of the immediate rhythm variable z_i but depends on the Bernoulli variable e_i to determine if it should connect up with the previous w_{i-1} to form a sequence. Thus with NLDA, explicit movement patterns are modelled. (b) Formal graphical model representation of NLDA in plate notation.

Realizing the need for richer contexts, in this paper, we extend our previous work [33], appealing to a richer Bayesian topic model known as n-gram topic model (NLDA) [40] for the task of rhythm extraction. Carrying

on from the example presented in Figure 3, the generative process and graphical model representation for NLDA is illustrated in Figure 4. As can be seen, NLDA is now explicitly capturing the arrival order of spatio-temporal words and is thus able to model temporal information.

The generative process for a word is slightly modified compared with that for LDA, described earlier. Each word w_{i+1} is now sampled conditionally on the previous word w_i and the current topic z . This is made explicit via a Bernoulli variable e_i : if e_i is true then w_{i+1} is sampled from a multinomial distribution $\text{Mult}(\gamma_{z,w_i})$ parameterised by both the current topic z and the previous word w_i ; otherwise w_{i+1} is sampled from another multinomial distribution $\text{Mult}(\phi_z)$ depending on the current topic z but independent of w_i . Intuitively, NLDA treats the arriving order of words at time i as a stochastic process: if e_i is true, two consecutive words w_i and w_{i+1} connect up to continue forming a longer movement pattern, otherwise a new movement pattern begins. Both LDA and NLDA are fully hierarchical Bayesian models, thus the parameters are further treated as random variables distributed according to its corresponding conjugate prior distribution, in our case Dirichlet distribution²: $e_i \sim \text{Bernoulli}(\epsilon)$, $\epsilon \sim \text{Beta}(\lambda)$, $\gamma_{zw} \sim \text{Dir}(\eta)$ and $\phi_z \sim \text{Dir}(\beta)$. In all cases, symmetric Dirichlet distribution is used, i.e., each distribution is parameterised by a single parameter.³ A topic or rhythm z in NLDA is now a distribution over n-grams parameterized $\phi_{z,w} = \Pr(w \mid z, e = 0)$ and $\gamma_{z,w} = \Pr(w_i = u \mid z, w_{i-1} = v, e = 1)$ which are subject to being inferred (cf. Algorithm 4).

Approximate inference with Gibbs sampling for LDA and NLDA. Exact inference in LDA and NLDA is known to be intractable. Options include the variational approach [5], expectation propagation (EP) [27] or collapsed Gibbs sampling [17]. Despite being deterministic with an analytical bound, the variational method is known to be biased and may wrongly estimate the parameter. EP requires memory storage in the order of number of topics \times total words in the corpus and quickly becomes infeasible with a large corpus. Besides, EP is known to have problems with sparse data. For LDA, we use collapsed Gibbs sampling, proposed in [17], which iteratively draws samples from the conditional distribution for each topic z_i after marginalizing out the parameters⁴:

$$\Pr(z_i^d = z \mid \mathbf{z}_{-i}, \mathbf{w}, \alpha, \beta) \propto (\alpha_z + n_z^{d,-i}) \frac{\beta_w + n_{zw}^{-i}}{\sum_v (\beta + n_{z,v}^{-i})} \quad (3)$$

where \mathbf{z}_{-i} denotes the sequence of topic assignments excluding position i (in document d), \mathbf{w} denotes the entire observed sequence of words, $n_z^{d,-i}$ denotes the number of topic z being assigned to document d excluding position i , $n_{z,w}^{-i}$ denotes the number of the current word $w = w_i^d$ being assigned to topic z , and $n_{z,v}^{-i}$ denotes the number of times a vocabulary word v being assigned to topic z , again excluding position i . The first term is proportional to the number of the current topic z within document d and the second term is proportional to the count of the current word w in document d to the topic z . Intuitively the effect of co-occurrence is achieved by assigning higher probability to two words in the same document being assigned to the same topic. The pseudo-code for rhythms extraction with LDA is presented in Algorithm 3, where we note that W is the size of vocabulary (i.e., the number of unique spatio-temporal words in the collection) and Equation (4) is the posterior estimate for $\phi_{z,w}$ conditional on the Gibbs topic assignment sample [17].

For NLDA, collapsed Gibbs sampling [40] involves integrating out the set of parameter variables $\{\theta, \phi, \gamma, \epsilon\}$, iteratively sampling from the following conditional distribution:

$$\Pr(z_i^d = k, e_i^d = c \mid \mathbf{z}_{-i}^d, \mathbf{e}_{-i}^d, \mathbf{w}, \alpha, \beta, \lambda, \eta) \propto \begin{cases} [\eta_c + p_{luc}] (\alpha_k + q_{dk}) \left[\frac{\beta_v + n_{kv}}{\beta_* + n_{k*}} \right] & \text{if } c = 0 \\ [\eta_c + p_{luc}] (\alpha_k + q_{dk}) \left[\frac{\lambda_v + m_{kv}}{\lambda_* + m_{k*}} \right] & \text{if } c = 1 \end{cases} \quad (5)$$

² where simply speaking, Bernoulli distribution is viewed as as a special case of Multinomial and Beta as a special of Dirichlet.

³ Readers are recommended to consult [5,17,40] for further explanation on LDA and NLDA, including their approaches to derive Gibbs sampling. In this paper, we only focus on the semantic of these models and key inference results rather than being mathematical rigor.

⁴ Please see [18] for full derivation.

Algorithm 3 Gibbs sampling for LDA.

Input: WiFi corpus $\{\mathbf{w}^1, \dots, \mathbf{w}^D\}$, hyper-parameters α, β , number of Gibbs samples L .

```
1: randomly initialize  $\mathbf{z}^d$  for all  $d = 1, \dots, D$ .
2: for  $s = 1$  to  $L$  do
3:   for  $d = 1$  to  $D$  do
4:     for  $i = 1$  to  $N_d$  do
5:       sample  $z_i^d$  according to Equation (3).
6:     end for
7:   end for
8:   store the  $s$ -th Gibbs sample.
9: end for
```

Output: estimated rhythms based on the probability $\Pr(w | z)$ using the last Gibbs sample:

$$\hat{\phi}_{z,w} = \frac{n_{z,w} + \beta}{\sum_{v=1}^W n_{z,v} + W\beta} \quad (4)$$

where p_{luc} is the count of observing the event $\{z_{i-1} = l, w_{i-1} = u, e_i = c\}$, q_{dk} is the count for topic k in document d , n_{kv} is the number of times the vocabulary word w is assigned to topic k , and m_{kuv} is the count for observing $\{z_{i-1} = k, w_{i-1} = u, w_i = v, e_i = 1\}$. The asterisk $*$ denotes the summing operation at the corresponding index, e.g., $n_{k*} = \sum_{v=1}^W n_{kv}$. In all cases, the current sampling position is always excluded during counting.

In Equation (5), the joint statistical strength of pattern $u \rightarrow w$ given topic k is captured by the count m_{kuv} . Loosely speaking, if we continue to observe a movement pattern of a worker, for example, from his office to the coffee machine and then to the meeting room every Tuesday, then the ‘count’ of this event increases, giving it a higher probability in Equation (5), and thus it is more likely to be detected during the inference process. The pseudo-code for rhythm extraction with NLDA is shown in Algorithm 4, where the posterior estimates in Equation (6) and (7) are used to derive the rhythms [40].

Algorithm 4 Gibbs sampling for NLDA.

Input: WiFi corpus $\{\mathbf{w}^1, \dots, \mathbf{w}^D\}$, hyper-parameters $\alpha, \beta, \eta, \lambda$, number of Gibbs samples L .

```
1: randomly initialize  $\mathbf{z}^d, \mathbf{e}^d$  for all  $d = 1, \dots, D$ .
2: for  $s = 1$  to  $L$  do
3:   for  $d = 1$  to  $D$  do
4:     for  $i = 1$  to  $N_d$  do
5:       sample  $z_i^d$  and  $e_i^d$  according to Equation (5).
6:     end for
7:   end for
8:   store the  $s$ -th Gibbs sample.
9: end for
```

Output: estimated rhythms based on the probability $\Pr(w | z)$:

$$\hat{\phi}_{z,w} = \frac{n_{z,w} + \beta}{\sum_{v=1}^W n_{z,v} + W\beta} \quad (6)$$

$$\hat{\gamma}_{z,uv} = \frac{m_{z,uv} + \eta}{\sum_{v=1}^W m_{z,uv} + W\eta} \quad (7)$$

Convergence and complexity. The collapsed Gibbs sampling presented for LDA and NLDA is a form of Markov Chain Monte Carlo method and is guaranteed to converge to the target distribution with sufficient number of iterations L [15]. These methods have been found to work very efficiently in practice when applied

to different types of corpora [17,18,40]. In this work, we employ a version of symmetric Dirichlet for both α and β which implies that the Dirichlet is controlled by only one parameter. As β gets smaller $\beta \rightarrow 0$, the model favours more discriminative topics. I.e., the samples generated from the Dirichlet distribution are concentrated at the corners of the simplex. When $\beta \rightarrow 1$ the Dirichlet behaves more like a uniform distribution, and when β is large, its samples are concentrated in the center of the simplex, favoring topics which are more similar (by Kullback-Leibler divergence). The hyperparameter λ in NLDA (cf. Figure 4) controls the sparsity among the movement patterns. Again, we use a small initialization for λ as we wish to favour longer movement patterns.

The complexity of deriving a sample from Equation (3) and (5) is $O(T)$ and $O(2T)$ respectively, with T being the number of topics in consideration. By inspection, the complexity of the Gibbs sampling routine in Algorithm (3) and (4) scales linearly with the number of words in the corpus $N = \sum_{d=1}^D N_d$ and linearly with the number of Gibbs samples L , giving the overall complexity of $O(TLN)$. In practice, T and L are often fixed in advance and thus, in our case, the term that really matters is the total number of words in the corpus N . In other words, our algorithms scale linearly with the number of WiFi observations acquired over time.

4. Experiments

4.1. Data and groundtruth

All WiFi data was collected using a handheld HP iPAQ HW6569 and custom logger written with .NET Compact Framework 2.0 and the free OpenNetCF package⁵. For motion classification task, groundtruth was manually marked down using a GUI interface (Fig. 5) as the user walked around the designated area. Data was collected over a 60 minute period at a sampling rate of $0.5Hz$ for 5 days. Figure 6b shows an example user trajectory, including static periods when the user was immobile at the same place for a few minutes. Data for the significant locations experiment was drawn using the same apparatus, but over a 28 day period of one user’s normal daily routines, during the hours of 8:00AM to 17:30PM. Groundtruth was labelled from among 6 landmarks indicated by the user to be significant, and, as with the motion groundtruth, was indicated using the logging software. For the rhythm experiment, 28 days of data collected for the significant locations experiment was collated and used.

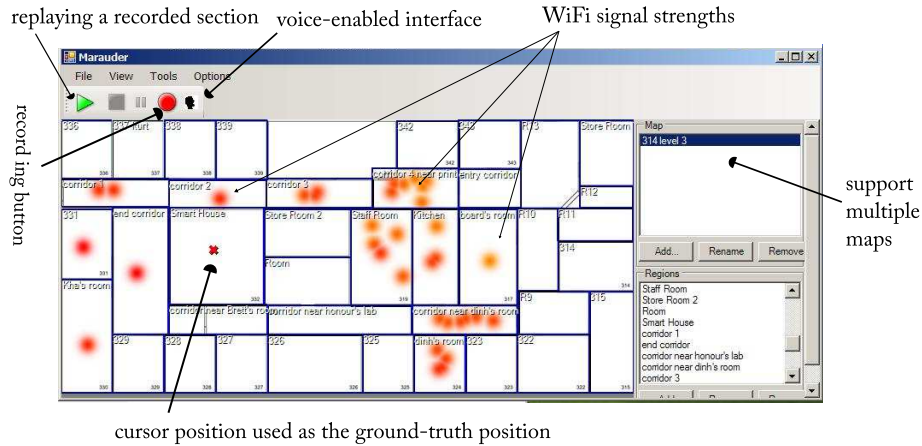
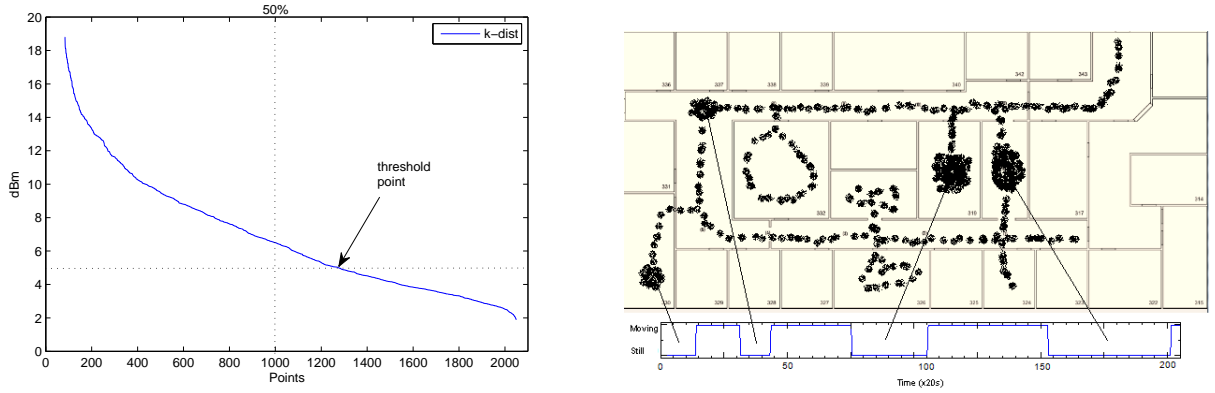


Figure 5. GUI for capturing WiFi signals.

⁵ www.opennetcf.com



(a) The sorted k -distance graph, $\delta = 90\%$, window size = 20s. (b) An example of user WiFi traces for motion classification task (groundtruth is plotted underneath).

Figure 6. k -sorted distance graph and example of WiFi trace.

4.2. Motion classification results

A number of different parametrizations were used to experiment on motion classification, including both overlapping and non-overlapping windows, and window sizes ranging from 10 to 120 seconds. The cluster quality threshold parameter δ was tested with 85%, 90% and 95% with a threshold of $\eta_0 = 0.75$ discussed in Eq.(2). We also compared our method with the Locadio method presented in Krumm et al. [23], which requires supervised training data to learn the probabilities $P(\sigma^2 \mid \text{still})$ and $P(\sigma^2 \mid \text{moving})$. Fig. 6 further plots the k -sorted distance graph to choose the threshold parameter used in the density-based clustering mentioned earlier and an example of WiFi trace collected over a day with marked position and its annotated motion state. To evaluate the algorithm, we compute the accuracy for each class defined as the ratio of the number of observations detected for that class to the total number of observations in testing.

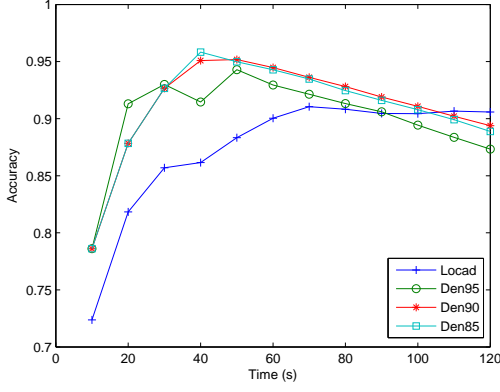
The accuracy of the Locadio of [23] and our density-based approaches are shown in Figures 7, experimented with and without overlapping window. It can be observed that the performance is poor for short windows (10s), probably due to the limited number of WiFi observations within the window. Accuracy gradually improves as the window is lengthened, leveling out subsequently. In general, the density-based approach shows superior performance compared to Locadio in all cases. Moreover, overlapping windows lead to more consistent performance. It is to be noted that while overlapping and non-overlapping windows yield more or less similar results, overlapping windows have shorter latency and thus can be used in real time.

4.3. Significant location extraction results

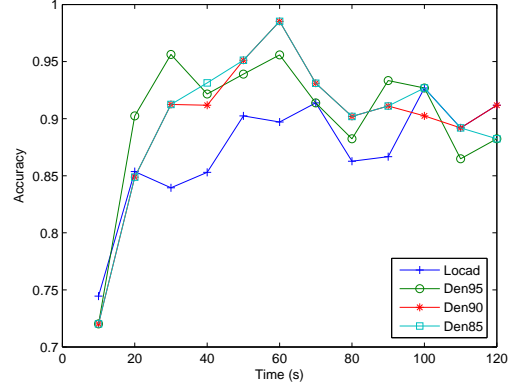
We first perform the motion classification step at the quality threshold of $\delta = 90\%$ to determine if the motion state within a non-overlap 60s-Window is static or not. Recall that a 60s-Window shows the best performance in non-overlap situation (Figure 7a). Two parameters in the batch algorithm are automatically derived, $\epsilon = 5\text{dBm}$ and $D = 60\text{s} \times 0.5\text{Hz} \times 90\% = 10$. We call this the “average observation” for a *minute interval*.

In the second step, we cluster this new average observation incrementally as described earlier. While the neighbor distance remains at $\epsilon = 5\text{dBm}$, the number of neighbors is configured to reflect a minimum duration spent a landmark, appropriate to discovery of significant locations. We empirically set $D = 5$. Note that the meaning of D in each step should be distinguished because of the different time intervals integrated in their observations.

For performance evaluation, we use the *cluster purity* measured as the percentage of match between the extracted clusters $\{c_1, \dots, c_L\}$ and the groundtruth clusters $\{\psi_1, \dots, \psi_K\}$. The purity of cluster i is defined as



(a) Result with overlapping window.



(b) Result with non-overlapping window.

Figure 7. Motion detection accuracy when the window size ranges from 10s to 120s in non-overlapping and overlapping cases. Legend consists of Locadio method in [23] (Locad) and our density-based approach with different density threshold values.

Landmark	#Groundtruth	#Clustered	Purity (%)
1	2574	2572	99.92
2	296	291	98.31
3	40	39	97.50
4	41	41	100.00
5	193	187	96.89
6	19	19	100.00

(a) Performance of clustering significant locations.

Landmark	#Prediction	#Correct	Accuracy(%)
1	857	855	99.76
2	100	94	94.00
3	13	13	100.00
4	14	13	92.85
5	35	31	88.57
6	7	5	71.42

(b) Prediction of significant locations.

Table 2
Significant detection results.

$$P_i = \frac{\max_j |c_j \cap \psi_i|}{N_i} \quad (8)$$

where N_i is the number of points in the groundtruth of cluster i . The average purity across K clusters is computed as:

$$\bar{P} = \sum_{i=1}^K \frac{N_i P_i}{N} \quad (9)$$

where N is total number of points $N = \sum_{i=1}^K N_i$.

The clusters are compared with the groundtruth and the purity of the clusters are shown in Table 2a. We achieved an overall cluster purity is 98.77% indicating that it is fairly robust against the WiFi noise. It is partially explained by the fact that the window size used for significant location detection is 60s, and thus, the number of WiFi signals collected are stable enough to make an accurate prediction.

To test the predictive ability of the significant location detection framework, one third of the collected data not used for the above clustering is used for testing. Again, incremental DBSCAN ($\epsilon = 5\text{dBm}$ and $D = 5$) is used to assign clusters, and thus significant locations. Results are shown in Table 2b. We achieve an average of more than 90% in accuracy, suggesting that it can also be used as a prediction model with reasonable performance. Furthermore, it appears from the results that when more data is available for (training) significant locations (Landmarks 1–5), the accuracy could be very high, almost perfect in some cases.

4.4. Rhythm detection results

Significant locations are discovered using the previous technique, and are mapped to words by discretizing each day into 30-minute intervals from 8am to 5:30pm to obtain a sequence of data consisting of pairs of $\langle \text{time, significant_place_label} \rangle$ tuple. We run rhythm detection algorithms with LDA and NLDA described in section 3.5. We also run the hidden Markov model (HMM) as the baseline performance for comparison.⁶

Briefly a HMM is parameterised with $\{\pi, A, B\}$ where π_i is the initial probability of state i at the first time slice, A_{ij} is the transition probability from state i to j and $B_{v|i}$ is the emission probability of symbol v given the state i . To train the HMM we set the number of states equal to the number of topics used in LDA and NLDA. The observation set consists of all observed time-stamped location in the training data set. We then run expectation maximization (EM) algorithm to estimate the HMM parameters with random initialization, setting the number of iterations to 50 with a stopping threshold of 10^{-5} in the log-likelihood difference between two consecutive EM iterations.⁷

4.4.1. Results with LDA

For LDA, Gibbs sampling described in Section 3.5 and Algorithm 3 are used with the Dirichlet hyper-parameters are set to small values to favor discriminative topics and document sparsity: $\alpha = 0.01$ and $\beta = 0.01$. Number of iterations is set to $L = 2000$ with 200 burn-in samples (discarded during the initial period of the MCMC chain) and subsequent samples are collected after every 10 samples – a strategy used in [17] which has shown good performance. Another input required for LDA is the number of topics – or equivalently, the number of distinct rhythms to be discovered. Again we use a strategy suggested in [17] for this model selection task by evaluating the goodness of the model measured by its perplexity in the following equation:

$$\text{Perp}(\mathbf{w}) = \exp \left\{ -\frac{\log \Pr(\mathbf{w})}{N} \right\} \quad (10)$$

where N is total number of words in the corpus and \mathbf{w} is the entire dataset. However, computing $\Pr(\mathbf{w})$ is non-trivial even with the Gibbs samples collected because the parameters θ and ϕ have been integrated. We adapt an important sampling approach used in [39] to compute $\Pr(\mathbf{w})$ as follows. After each sampling step s , let $n_{d,k}^{(s)}$ be the number of times that topic k appears in document d , $n_{k,v}^{(s)}$ be the number of times a word v is assigned to topic k , parameters $\theta^{(s)}$ and $\phi^{(s)}$ for this sample are first estimated:

$$\hat{\theta}_{d,k}^{(s)} = \frac{n_{d,k}^{(s)} + \alpha}{\sum_{k=1}^T n_{d,k}^{(s)} + T\alpha} \quad \hat{\phi}_{k,v}^{(s)} = \frac{n_{k,v}^{(s)} + \beta}{\sum_{v=1}^W n_{k,v}^{(s)} + W\beta} \quad (11)$$

and then $\Pr(\mathbf{w})$ is given as:

$$\Pr(\mathbf{w}) = \prod_{d=1}^D \prod_{i=1}^{N_d} \sum_{k=1}^T \frac{1}{S} \sum_{s=1}^S \hat{\theta}_{d,k}^{(s)} \hat{\phi}_{k,w_i^d}^{(s)} \quad (12)$$

where S is the number of collected samples. The pseudo-code for this procedure is shown in Algorithm 5.

Figure 8 shows the perplexity curve w.r.t the number of topics. The lower the perplexity, the better the model fits. In general, the model perplexity decreases as the number of topics increases. However, the greater the number of topics, the more likely the model over explains the data and the more sampling computation and storage required. The general rule of thumb is to choose a balance between simplicity of the model and the degree of fitness. In our case (cf. Figure 8) we empirically choose the number of topics $T = 5$ where the perplexity seems to decrease rapidly and appear to settle down.

⁶ Basic familiarity with the HMM is assumed, e.g., see [35].

⁷ Knowing that this EM procedure is sensitive to initialization, we have made several EM runs. The obtained results are, however, rather consistent in the estimated emission probability matrix and the Viterbi decoding results, and thus, we report in this paper one of this results.

Algorithm 5 Selecting number of topics in LDA based on perplexity.

- 1: **for** $T = 1$ to a fixed number of topics (e.g., 20) **do**
 - 2: run Gibbs sampling routine for LDA in Algorithm 3 with the number of topics set to T .
 - 3: collect S Gibbs samples from above procedure.
 - 4: **for** $s = 1$ to S **do**
 - 5: compute $\hat{\theta}^{(s)}$ and $\hat{\phi}^{(s)}$ from Equation (11).
 - 6: **end for**
 - 7: compute the perplexity wrt T from Equation (12).
 - 8: **end for**
 - 9: Output the optimal value of T according the perplexity computed in step 7.
-

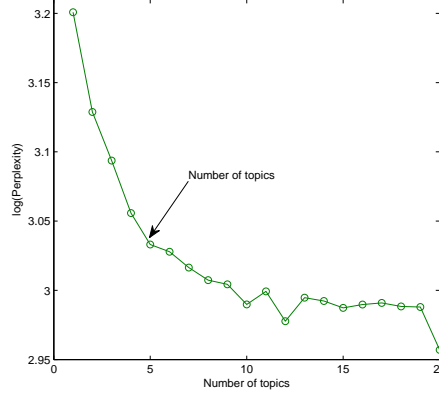
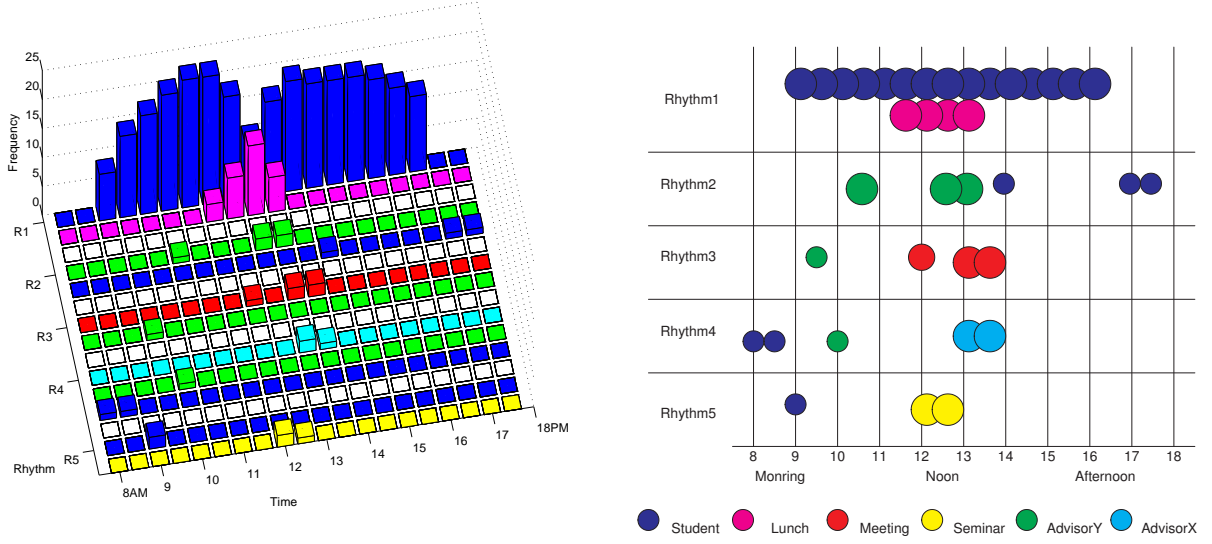


Figure 8. Choosing number of topics using perplexity.

With the chosen $T = 5$, we interpret the results from LDA by examining the probability $\hat{\phi}_{k,v}$ of a word w given a topic k . In other words, we look at those time-stamped locations that are assigned to a rhythm, interpreted as topic k , with high probability. Figure 9 shows five rhythms in which each row corresponds to a specific landmark and the height of each bar depicts the frequency of landmark accumulated over all collected Gibbs samples. Figure 9b provides a different perspective on the same detected rhythms where the top landmarks (with high probability) within each rhythms are plotted where the time information is recovered for easy visual interpretation (e.g., if the word is '1200lunch' then we plot a label 'lunch' at 12:00pm). Note that there is no sequential ordering information results from this LDA inference.

A general observation is that the LDA has learned reasonably well the repetitive office patterns of a research student. For example, the first rhythms corresponds to his typical daily routines of being in the office and having lunch. This person participates in the institute's academic activities, such as academic meeting (Rhythm 3) and attending seminars (Rhythm 5). In addition, his frequent visits to two advisor's rooms for discussions are also extracted well clustered (Rhythm 2 and Rhythm 4).

For the HMM, the probability of a time-stamped landmark given a rhythm is extracted from the learned observation emission matrix $B_{v|k}$. In general, we do not observe good grouping across the HMM states. State 2 seems to correspond more to event of meeting with AdvisorY (this becomes more evident in the Viterbi decoding results shown in Figure A.3 presented in the next section). State 3 tends to account for the afternoon student routines inside his office (again, more evidently in Figure A.3). In contrast, LDA grouping gives better and more meaningful interpretation.



(a) Five rhythms detected as five topics from LDA, each rhythms are delineated by white boxes (best viewed in color) (b) Projected view of detected rhythms as mixture of landmarks (sorted according to time).

Figure 9. Rhythms detection results with LDA.

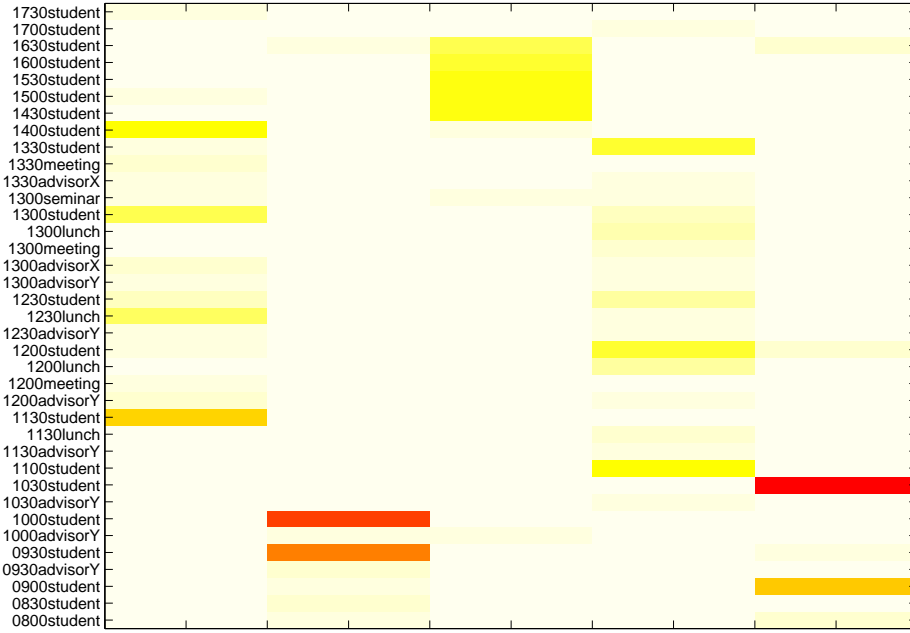


Figure 10. Learned probability distributions over landmarks for each HMM state, plotted according to the emission probability matrix $B_{v|i}$ from the trained HMM.

4.4.2. Results with NLDA

While LDA has shown good rhythms detection, its shortcomings is the loss of temporal ordering between landmarks since it treats them exchangeably within a day – a popular assumption known as bag-of-word. Its n-gram extension NLDA presented in section 3.5 is to overcome this problem, allow us to extract rhythms with real temporal patterns discovered automatically from data. Again, we run NLDA with the number of

topics⁸ set to 5, hyper-parameters⁹ $\alpha = 0.1, \beta = 0.01, \lambda = 0.01$ and $\eta = 0.01$.¹⁰ We interpret the Gibbs samples to discover the n-gram by extracting and accumulating the temporal extent where e_i is true and the topic is unchanged. This basically forms a movement pattern detected that topic where, again, we interpret topics as rhythms. The frequency of how often a movement pattern detected for that topic, after normalized becomes its probability assigned to the topic.

For the HMM, using the trained model, we run Viterbi decoding algorithm on the training dataset and the patterns detected for each state is similar to the NLDA case, i.e., if we observe n -consecutive decoded states to be the same, the corresponding segment of observations become a movement pattern for that states. The results for NLDA are shown in Figure 11 and HMM are shown in the appendix in Figure A.3.

The results from NLDA are rather interesting. Four of five rhythms have strong interpretation as shown in Figure 11. We can see that the first rhythm detects a kind of typical daily patterns of the student working by himself in the office. The second rhythm tends to capture typical ‘meeting’ patterns of the student where he goes from his office to the lunch room, and then to various meeting places (AdvisorX, meeting room, AdvisorY), and finally return to his office. Rhythm 3 is loosely attached with going to seminar. Finally, it is interesting to see that Rhythm 4 has captured various movement patterns between the student office and the AdvisorY. This is perhaps because during the initial period of capturing, this student has met with AdvisorY intensively at different time of the day (where in contrast to AdvisorX, he only meets at either 1:00pm or 1:30pm). A full illustration of patterns within each rhythms is further provided in Figure A.1 and A.2 in the appendix.

The results from HMM decodings shows that it tends to over-segment the data, resulting into much smaller and less useful patterns. It is also evident that the HMM has also learned certain meaningful rhythms but its results are much less interpretable. For example state 2 of the HMM in A.3 shows that it also captures the ‘meeting AdvisorY’ routines, analogous to rhythm 4 of NLDA, but its discovered movement patterns are much less shorter. Likewise, state 1 captures a set of different landmarks and has failed to capture the sequential information (e.g, where is the student before he come to meeting room).

5. Conclusion

Motivated by the need to build intelligent context-aware systems (e.g., assistive technology for the visually impaired, push content information system) in indoor environments, we have presented novel, unsupervised clustering method for the extraction of user context from ambient WiFi, continuing from our previous work in [33]. This includes state of motion, significant locations, and rhythms. While most of existing work found in the literature has used WiFi signals for the purpose of localization and often require heavy calibration and supervised training, our work has departed from a different angle, utilizing WiFi signals from a completely unsupervised and incremental approach. In addition to the novel algorithms for high accuracy motion and significant places detection in real-time using WiFi signals presented in [33], we have also designed and developed a new approach to extract high-order movement patterns termed as rhythms using Bayesian probabilistic models in this paper. Experimental results, in comparison with existing methods, suggest that our approaches are fast and robust against the sparsity nature of WiFi signals. Interesting, consistent and interpretable rhythms are also automatically derived, suggesting richer contexts capturing spatio-temporal information can also be inferred directly from raw WiFi signals. The advantages of our approach lie in its use of existing wireless infrastructure, without requirements for calibration, and its ability to support real-time services. In addition, being able to infer the context of the mobile user is a vital, foundational

⁸ We would like to note that we can use the same perplexity-based model selection procedure for LDA here; however, we set the number of topics to 5 for consistency across all models, including the HMM.

⁹ Note that these hyper-parameters can also be estimated by maximizing the type-II likelihood (marginal likelihood) using Monte Carlo EM [15]; however it is beyond the scope of this paper.

¹⁰ We experimented with two strategies suggested in [40] where we do and do not force $z_{i+1} = z_i$ when $e_i = 1$. However, we found that the former gives much more meaningful patterns than the later. This is opposite with what [40] reported in their work when they applied NLDA for text data with retrieval evaluation. This is an interesting observation in our view. A probable explanation is because the nature of data and tasks experimented in [40] ours are quite different: one is to do with a huge dataset of texts for retrieval task, ours deal with smaller non-linguistic data and for the purpose of pattern discovery.

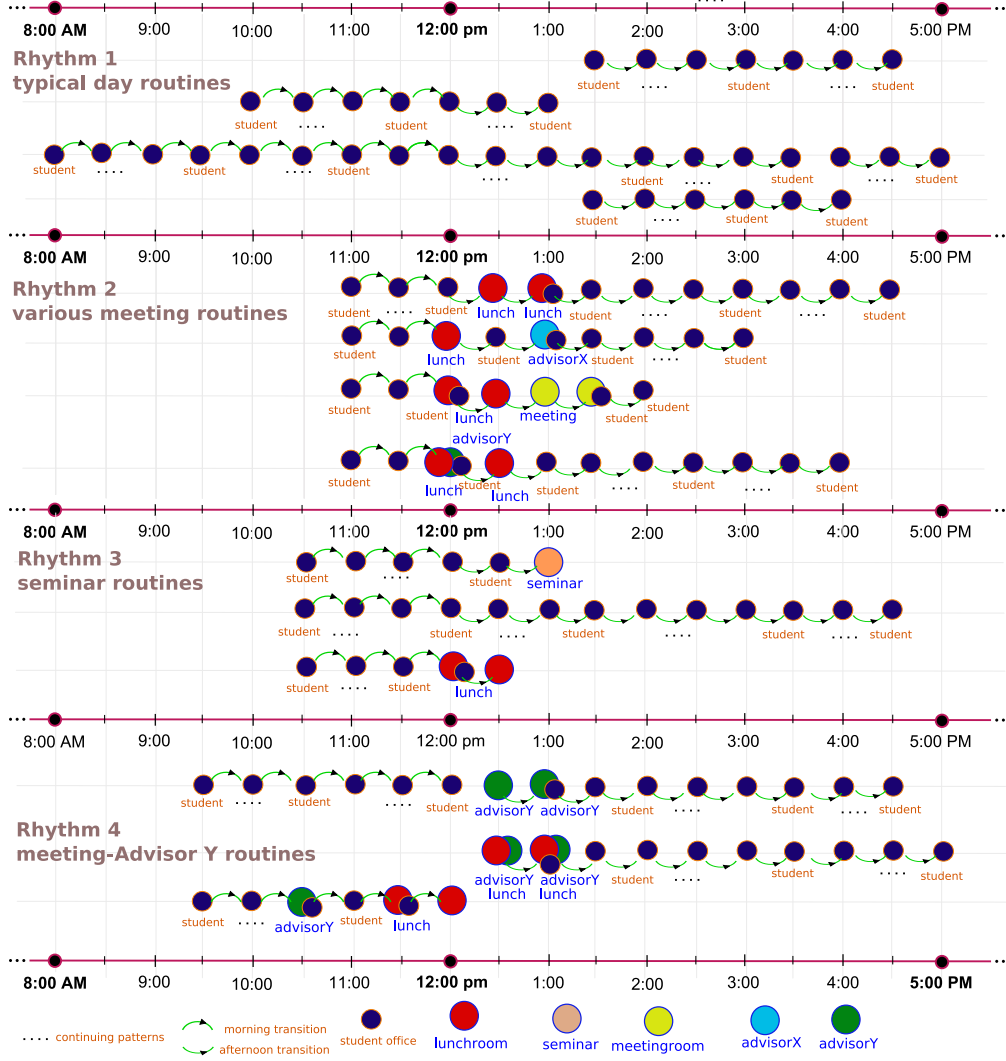


Figure 11. Top patterns within each rhythms detected with NLDA (comparison with HMM results are shown in Figure A.3).

component in pervasive computing. We have presented methods in this regard which can serve as a basis for both annotation and prediction at a number of levels of the services stack such as context-sensitive device resource and interface management, personalized push-information; and in a shared context, it can aid market research, surveillance and urban planning.

Acknowledgment

We would like to acknowledge the work of Kha Tran during the initial period of this work. His help with data collection and some programming tasks are gratefully acknowledged. We also thank the anonymous reviewers for their constructive comments to improve the quality of the paper.

Appendix A. All patterns discovered with NLDA

Figure A.1 and A.2 further shows all movement patterns with rhythms detected by NLDA. The former shows the first two rhythms and the later shows the next three rhythms. Figure A.3 shows all movement

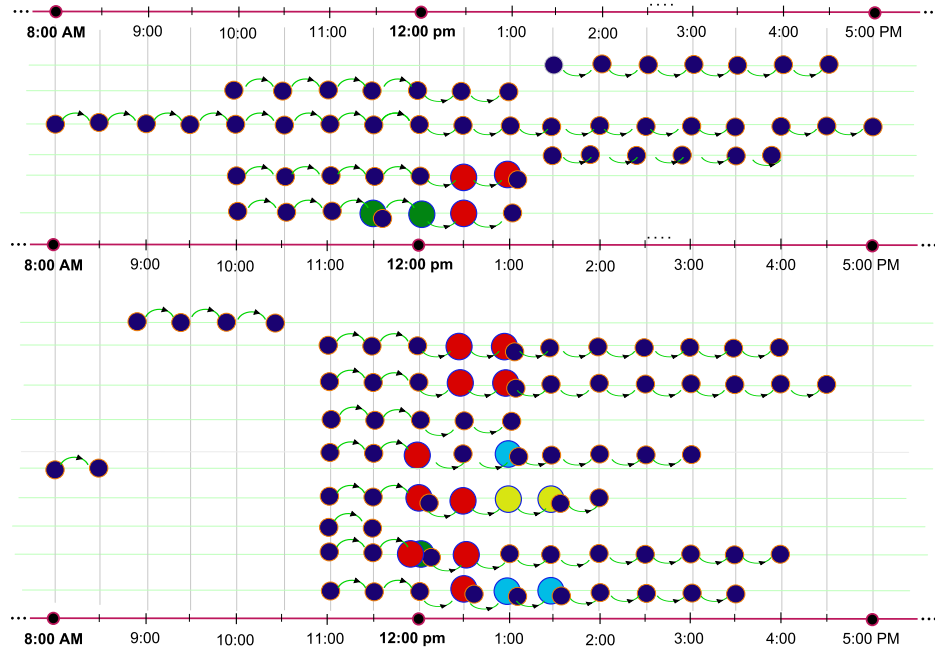


Figure A.1. All movement patterns detected by NLDA for the first two rhythms (best viewed in color).

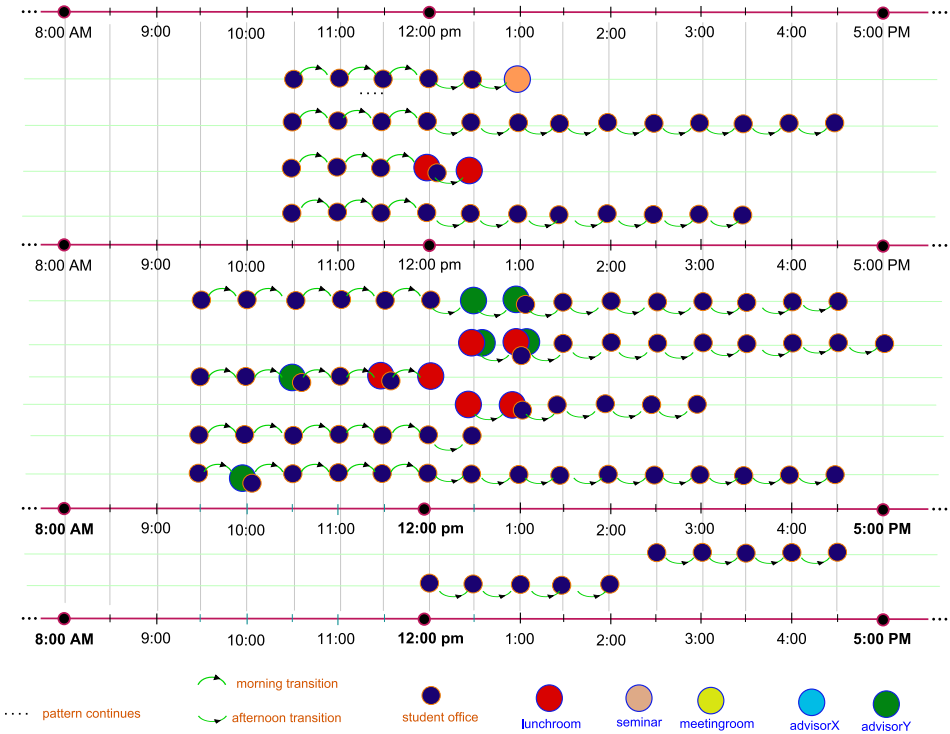


Figure A.2. All movement patterns detected by NLDA for the next three rhythms (best viewed in color).

patterns detected by HMM using Viterbi decoding. It is visually clear that HMM returns many more short and spurious movement patterns than NLDA, indicating poorer structures were extracted.

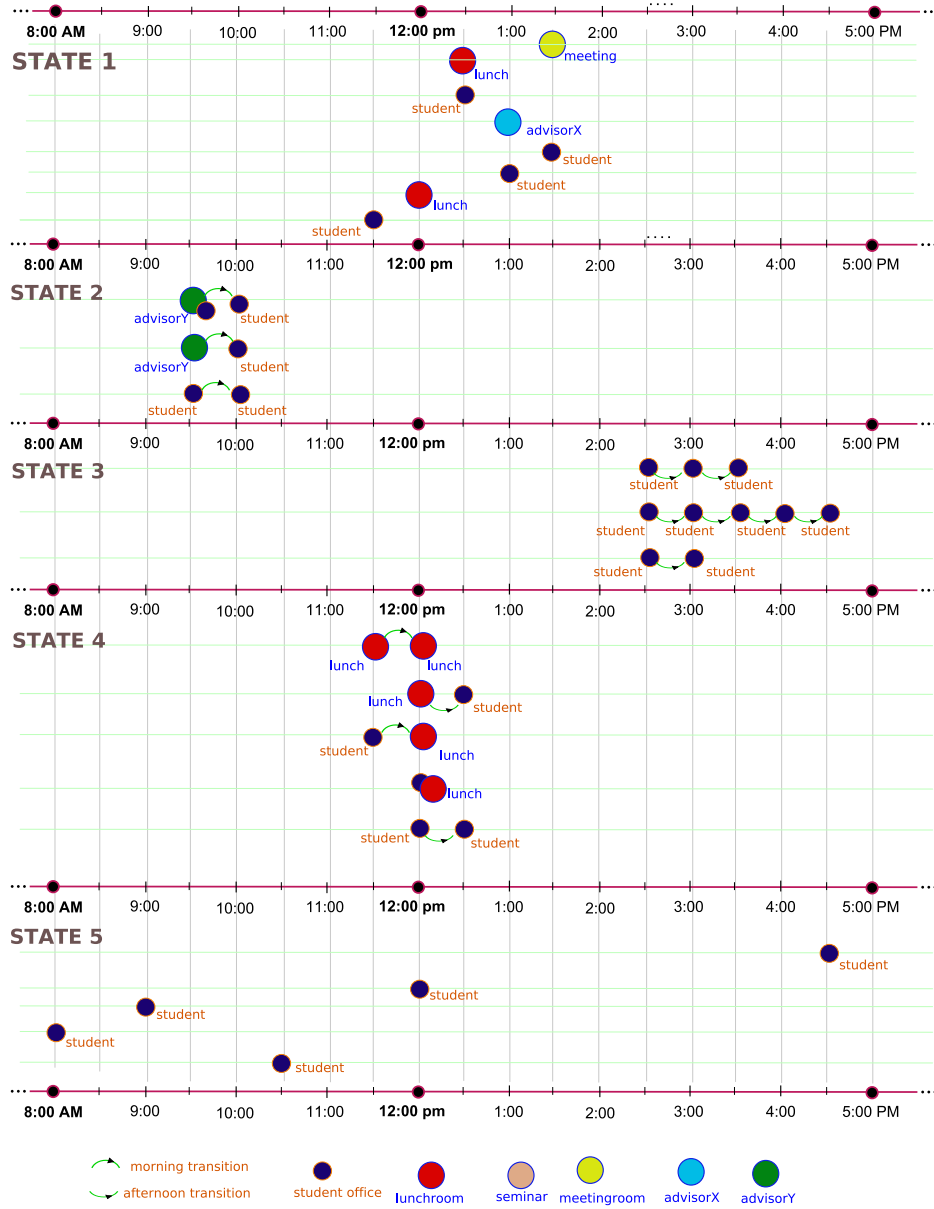


Figure A.3. All movement patterns detected with HMM Viterbi decoding.

References

- [1] B. Adams, D. Phung, and S. Venkatesh. Extraction of social context and application to personal multimedia exploration. *Proc. ACM Int. Conf. on Multimedia (ACM-MM)*, pages 987–996, 2006.
- [2] B. Adams, D. Phung, and S. Venkatesh. Sensing and using social context. *ACM Trans. on Multimedia Computing, Communications and Applications*, 5(2):1–27, 2008.
- [3] D. Ashbrook and T. Starner. Using GPS to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing*, 7(5):275–286, 2003.
- [4] P. Bahl and VN Padmanabhan. RADAR: an in-building RF-based user location and tracking system. In *Proc. Annual Joint Conf. of the IEEE Computer and Communications Societies*, volume 2, 2000.
- [5] David M. Blei, Andrew Y. Ng., and Michael I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
- [6] M. Blum. Real-time context recognition. Master's thesis, Department of Information Technology and Electrical Engineering, Swiss Federal Institute of Technology Zurich, 2005.

- [7] H.H. Bui, S. Venkatesh, and G. West. Policy recognition in the abstract hidden Markov model. *Journal of Artificial Intelligence Research*, 17(45):1–499, 2002.
- [8] J. Carrasco and E. Miller. Exploring the propensity to perform social activities: a social network approach. *Transportation*, 33(5):463–480, September 2006.
- [9] B.P. Clarkson. *Life patterns: structure from wearable sensors*. PhD thesis, Massachusetts Institute of Technology, 2003.
- [10] J. Donald, L. Lin, and F. Dieter. Inferring high-level behavior from low-level sensors. In *Proc. Int. Conf. on Ubiquitous Computing (UbiComp)*, volume 139, 2003.
- [11] N.N. Eagle. *Machine Perception and Learning of Complex Social Systems*. PhD thesis, Massachusetts Institute of Technology, 2005.
- [12] M. Ester, Hans P. Kriegel, Jorg Sander, Michael Wimmer, and Xiaowei Xu. Incremental clustering for mining in a data warehousing environment. In *Proc. Int. Conf. on Very Large Databases*, pages 323–333, 1998.
- [13] M. Ester, H.P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. Int. Conf. on Knowledge Discovery and Data Mining (ICDM)*, pages 226–231. AAAI Press, 1996.
- [14] V. Ferrari and D. Gatica-Perez. Discovering human routines from cell phone data with topic models. In *Proc. IEEE Int. Symp. on Wearable Computers (ISWC)*, September, Pittsburgh, Pennsylvania 2008.
- [15] W.R. Gilks, S. Richardson, and DJ Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC, 1996.
- [16] M.C. Gonzalez, C.A. Hidalgo, and A.-L. Barabási. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, June 2008.
- [17] T.L. Griffiths. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235, 2004.
- [18] T.L. Griffiths, M. Steyvers, and J.B. Tenenbaum. Topics in semantic representation. *Psychological Review*, 114(2):211–44, 2007.
- [19] R. Hariharan and K. Toyama. *Project Lachesis: Parsing and Modeling Location Histories*, volume 3234 of *Lecture Notes in Computer Science*, pages 106–124. 2004.
- [20] K. Henriksen, J. Indulska, and A. Rakotonirainy. *Modeling context information in pervasive computing systems*, pages 167–180. Lecture notes in computer science. Springer, 2002.
- [21] T. Huynh, M. Fritz, and B. Schiele. Discovery of activity patterns using topic models. In *Proc. Int. Conf. on Ubiquitous Computing (UbiComp)*. Springer, 2008.
- [22] Q. Jones, S.A. Grandhi, S. Whittaker, K. Chivakula, and L. Terveen. Putting systems into place: a qualitative study of design requirements for location-aware community systems. In *Proc. ACM Int. Conf. on Computer Supported Cooperative Work*, pages 202–211, New York, NY, USA, 2004. ACM Press.
- [23] J. Krumm and E. Horvitz. Locadio: Inferring motion and location from WiFi signal strengths. In *Proc. Int. Conf. on Mobile and Ubiquitous Systems: Networking and Services*, 2004.
- [24] A.M. Ladd, K.E. Bekris, A. Rudys, L.E. Kavraki, and D.S. Wallach. Robotics-based location sensing using wireless ethernet. In *Proc. ACM Int. Conf. on Mobile Computing and Networking*, 2002.
- [25] L. Liao, D. Fox, and H. Kautz. Extracting places and activities from GPS traces using hierarchical conditional random fields. *The International Journal of Robotics Research*, 26(1):119, 2007.
- [26] J.-L. Meunier. Peer-to-peer determination of proximity using wireless network data. In *Proc. IEEE Conf. on Pervasive Computing and Communications Workshops*, page 70, Los Alamitos, CA, USA, 2004. IEEE Computer Society.
- [27] T.P. Minka. Expectation propagation for approximate Bayesian inference. In *Proc. Conf. on Uncertainty in Artificial Intelligence (UAI)*, volume 17, pages 362–369, 2001.
- [28] Nam T Nguyen, Dinh Phung, H. H. Bui, and S. Venkatesh. Learning and detecting activities from movement trajectories using the hierarchical hidden markov model. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, volume 1, pages 955–960, San Diego, 2005. IEEE Computer Society.
- [29] D. Nicklas, M. Grossmann, J. Mínguez, and M. Wieland. Adding high-level reasoning to efficient low-level context management: A hybrid approach. In *Proc. IEEE Int. Conf. on Pervasive Computing and Communications (PerCom)*, pages 447–452. IEEE Computer Society Washington, DC, USA, 2008.
- [30] P. Nurmi and J. Koolwaaij. Identifying meaningful locations. In *Proc. Int. Conf. on Mobile and Ubiquitous Systems: Networking and Services*, pages 1–8, 2006.
- [31] S.N. Patel, K.N. Truong, and G.D. Abowd. *Powerline positioning: A practical sub-room-level indoor location system for domestic use*, volume 4206 of *Lecture Notes in Computer Science*, page 441. Springer, 2006.
- [32] M. Philipose, K.P. Fishkin, M. Perkowitz, D.J. Patterson, D. Fox, H. Kautz, and D. Hähnel. Inferring activities from interactions with objects. *IEEE Pervasive Computing*, pages 50–57, 2004.
- [33] Dinh Phung, Brett Adams, Kha Tran, Svetha Venkatesh, and Mohan Kumar. High accuracy context recovery using clustering mechanisms. In *Proc. IEEE Int. Conf. on Pervasive Computing and Communications (PerCom)*, pages 1–9, Los Alamitos, CA, USA, 2009. IEEE Computer Society.
- [34] Dinh Phung, Brett Adams, and Svetha Venkatesh. Computable social patterns from sparse sensor data. In *Proc. First Int. Workshop on Location Web, World Wide Web Conference (WWW)*, pages 69–72, New York, NY, USA, 2008. ACM.
- [35] LR Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
- [36] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen. A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks*, 9(3):155–164, 2002.
- [37] N. Roy, A. Roy, and S.K. Das. Context-aware resource management in multi-inhabitant smart homes: A Nash H-learning based approach. In *Proc. IEEE Int. Conf. on Pervasive Computing and Communications (PerCom)*, 2006.

- [38] V. Seshadri, G.V. Zaruba, and M. Huber. A Bayesian sampling approach to indoor localization of wireless devices using received signal strength indication. In *Proc. IEEE Int. Conf. on Pervasive Computing and Communications (PerCom)*, pages 75–84, 2005.
- [39] Y.W. Teh, D. Newman, and M. Welling. A collapsed variational Bayesian inference algorithm for latent Dirichlet allocation. In *Advances in Neural Information Processing Systems (NIPS)*, volume 19, 2007.
- [40] X. Wang, A. McCallum, and X. Wei. Topical N-grams: Phrase and topic discovery, with an application to information retrieval. In *Proc. IEEE Int. Conf. on Data Mining (ICDM)*, pages 697–702, 2007.
- [41] M. Weiser. The computer for the 21st century. *ACM Mobile Computing and Communications Review*, 3(3):3–11, 1999.
- [42] Z. Xiang, S. Song, J. Chen, H. Wang, J. Huang, and X. Gao. A wireless LAN-based indoor positioning technology. *IBM Journal of Research and Development*, 48(5/6):617–626, 2004.
- [43] J. Yin, X. Chai, and Q. Yang. High-level goal recognition in a wireless LAN. In *Proc. Nat. Conf. on Artificial Intelligence (AAAI)*, pages 578–584, AAAI Press 2004.
- [44] MA Youssef, A. Agrawala, and A. Udaya Shankar. WLAN location determination via clustering and probability distributions. In *Proc. IEEE Int. Conf. on Pervasive Computing and Communications (PerCom)*, pages 143–150, 2003.
- [45] C. Zhou, S. Shekhar, and L. Terveen. Discovering personal paths from sparse GPS traces. In *Proc. First Int. Workshop on Data Mining, Joint Conf. on Information Sciences*, 2005.